The Block Generative Topographic Mapping

Rodolphe Priam[1], Mohamed Nadif[2], and Gérard Govaert[3]

[1] LMA Poitiers UMR 6086, Université de Poitiers,
86962 Futuroscope Chasseneuil, France
[2] CRIP5 EA N°2517, Université Paris Descartes,
UFR de Mathématiques et Informatique, 75006 Paris, France
[3] Heudiasyc UMR 6599, UTC, BP 20529, 60205 Compiègne, France

Abstract. This paper presents a generative model and its estimation allowing to visualize binary data. Our approach is based on the Bernoulli block mixture model and the probabilistic self-organizing maps. This leads to an efficient variant of Generative Topographic Mapping. The obtained method is parsimonious and relevant on real data.

1 Introduction

Linear methods for exploratory visualization [1] are very powerful and contribute effectively to data analysis every days, but large datasets require new efficient methods. Indeed, the algorithms based on the matricial decomposition become useless for large matrices; moreover, the construction of many maps due to high-dimensionality makes the task of interpretation difficult from the information disseminated on the different maps. Finally a great quantity of data implies a great quantity of information to be synthesized and complex relations between individuals and studied variables. It is then relevant, in this context, to use a self-organizing map (SOM) of Kohonen [2]. SOM is a clustering method with a vicinity constraint on the cluster centers to give a topological sense to the obtained final partition. The SOM can be seen like an alternative of the k-means algorithm integrating a topological constraint on the centers. Bishop et al. [3] has re-formulated SOM within a probabilistic setting to give the Generative Topographic Mapping (GTM). GTM is a method similar to the self-organizing map with constraints of vicinity embedded in a mixture model of gaussian densities. In contrast to SOM, GTM is based on a well-defined criterion; the model implements an EM algorithm [4] which guarantees the convergence. Recently, to tackle the visualization of binary data, we have proposed a variant of GTM based on the classical Bernoulli mixture model [5]. The obtained results are encouraging but when the number of parameters increases with the high-dimensional data, the projection is therefore problematic. To cope with this problem, we propose in this work to use a parsimonious model in order to overcome the high-dimensionality problem.

When the data matrix $\mathbf{x}$ is defined on a set I of objects (rows, observations) and a set J of variables (columns, attributes), the block clustering methods, in contrast to the classical clustering methods, consider the two sets I and J

L. Prevost, S. Marinai, and F. Schwenker (Eds.): ANNPR 2008, LNAI 5064, pp. 13–23, 2008.

simultaneously [6],[7,8],[9]. Recently, these kind of methods were embedded in the mixture approach [10],[11],[12] and a parsimonious model called *Block Latent Model* has been proposed [13,14]. The developed hard and soft algorithms appeared more profitable than the clustering applied separately on I and J [15]. For these reasons, we propose to tackle the problem of visualization of I, by combining the block mixture model and the probabilistic self-organizing maps. This leads to propose a new generative topographic model.

This paper is organized as follows. In Section 2, to give the necessary background of the block clustering approach under the mixture model, we review the block latent model. In Section 3, we focus on the binary data and we propose a *Block Generative Topographic Mapping* based on a block Bernoulli model. In Section 4 devoted to the numerical experiments, we illustrate our method with three binary benchmarks. Finally, the last section summarizes the main points of work and indicates some perspectives.

Hereafter, the partition $\mathbf{z}$ into g clusters of a sample I will be represented by the classification matrix $(z_{ik}, i = 1, \ldots, n, k = 1, \ldots, g)$ where $z_{ik} = 1$ if i belongs to cluster k and 0 otherwise. A similar notation will be used for a partition $\mathbf{w}$ into m clusters of the set J. Moreover, to simplify the notation, the sums and the products relating to rows, columns, row clusters and column clusters will be subscripted respectively by the letters i, j, k and ℓ, without indicating the limits of variation which will be implicit. So, for example, the sum $\sum_i$ stands for $\sum_{i=1}^{n}$, and $\sum_{i,j,k,\ell}$ stands for $\sum_{i=1}^{n}\sum_{j=1}^{d}\sum_{k=1}^{g}\sum_{\ell=1}^{m}$.

2 The Latent Block Model

2.1 Block Clustering

In the following, the $n \times d$ matrix data is defined by $\mathbf{x} = \{(x_{ij}); i \in I \text{ and } \in J\}$ where $x_{ij} \in \{0, 1\}$. The aim of block clustering is to try to summary this matrix by homogeneous blocks. This problem can be studied under the simultaneous partition approach of two sets I and J into g and m clusters respectively. Govaert [7,8] has proposed several algorithms which perform block clustering on contingency tables, binary, continuous and categorical data. These algorithms consist in optimizing a criterion $E(\mathbf{z}, \mathbf{w}, \mathbf{a})$, where $\mathbf{z}$ is a partition of I into g clusters, $\mathbf{w}$ is a partition of J into m clusters and $\mathbf{a}$ is a $g \times m$ matrix which can be viewed as a summary of the data matrix $\mathbf{x}$. A more precise definition of this summary and criterion E will depend on the nature of data. The search of the optimal partitions $\mathbf{z}$ and $\mathbf{w}$ was made using an iterative algorithm. This one is based on the alternated k-means with appropriate metric applied on reduced intermediate $g \times d$ and $n \times m$ matrices. In [13,14], these methods were modeled in the mixture approach. Hard and soft algorithms were then developed. Efficient and scalability are the advantages of these new methods. Next, we review this approach.

2.2 Definition of the Model

Some of the most popular heuristic clustering methods can be viewed as approximate estimations of probability models. For instance, the inertia criterion optimized by the k-means algorithm corresponds to the hypothesis of a population arising from a gaussian mixture. For the classical mixture model, the probability density function (pdf) of a mixture sample $\mathbf{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ can be also written [13] $f(\mathbf{x};\boldsymbol{\theta}) = \sum_{\mathbf{z}\in\mathcal{Z}} p(\mathbf{z};\boldsymbol{\theta}) f(\mathbf{x}|\mathbf{z};\boldsymbol{\theta})$ where $\mathcal{Z}$ denotes the set of all possible assignments $\mathbf{z}$ of I into g clusters, $p(\mathbf{z};\boldsymbol{\theta}) = \prod_{i,k} p_k^{z_{ik}}$ and $f(\mathbf{x}|\mathbf{z};\boldsymbol{\theta}) = \prod_{i,k} \varphi(x_i;\alpha_k)^{z_{ik}}$. In the context of the block clustering problem, this formulation can be extended to propose a latent block model defined by the following pdf $f(\mathbf{x};\boldsymbol{\theta}) = \sum_{\mathbf{u}\in U} p(\mathbf{u};\boldsymbol{\theta}) f(\mathbf{x}|\mathbf{u};\boldsymbol{\theta})$ where U denotes the set of all possible assignments of $I \times J$, and $\boldsymbol{\theta}$ is the parameter of this mixture model.

In restricting this model to a set of assignments of $I \times J$ defined by a product of assignments of I and J, assumed to be independent, we obtain the following decomposition

$$f(\mathbf{x};\boldsymbol{\theta}) = \sum_{(\mathbf{z},\mathbf{w})\in\mathcal{Z}\times\mathcal{W}} p(\mathbf{z};\boldsymbol{\theta}) p(\mathbf{w};\boldsymbol{\theta}) f(\mathbf{x}|\mathbf{z},\mathbf{w};\boldsymbol{\theta}),$$

where $\mathcal{Z}$ and $\mathcal{W}$ denote the sets of all possible assignments $\mathbf{z}$ of I and $\mathbf{w}$ of J. Now, as in latent class analysis, the $n \times d$ random variables generating the observed x_{ij} cells are assumed to be independent once $\mathbf{z}$ and $\mathbf{w}$ are fixed; we then have

$$f(\mathbf{x}|\mathbf{z},\mathbf{w};\boldsymbol{\theta}) = \prod_{i,j,k,\ell} \varphi(x_{ij};\alpha_{k\ell})^{z_{ik}w_{j\ell}},$$

where $\varphi(.;\alpha_{k\ell})$ is a pdf defined on the real set $\mathbb{R}$ and $\alpha_{k\ell}$ an unknown parameter. The parameter $\boldsymbol{\theta}$ is formed by $\boldsymbol{\alpha} = (\alpha_{11}, \ldots, \alpha_{gm})$, $\mathbf{p}$ and $\mathbf{q}$; $\mathbf{p} = (p_1, \ldots, p_g)$ and $\mathbf{q} = (q_1, \ldots, q_m)$ are the vectors of probabilities p_k and q_ℓ that a row and a column belong to the kth component and to the ℓth component respectively.

For instance, for binary data, we obtain a Bernoulli latent block model defined by the following pdf

$$f(\mathbf{x};\boldsymbol{\theta}) = \sum_{(\mathbf{z},\mathbf{w})\in\mathcal{Z}\times\mathcal{W}} \prod_{i,k} p_k^{z_{ik}} \prod_{j,\ell} q_\ell^{w_{j\ell}} \prod_{i,j,k,\ell} (\alpha_{k\ell})^{x_{ij}} (1-\alpha_{k\ell})^{1-x_{ij}},$$

where $x_{ij} \in \{0,1\}$, and $\alpha_{k\ell} \in (0,1)$. Using this block model is dramatically more parsimonious than using a classical mixture model on each set I and J: for instance, with $n = 1000$ objects and $d = 500$ variables and equal class probabilities $p_k = 1/g$ and $q_\ell = 1/m$, if we need to cluster the binary data matrix into $g = 4$ clusters of rows and $m = 3$ clusters of columns, the Bernoulli latent block model will involve the estimation of 12 parameters $\boldsymbol{\alpha} = (\alpha_{k\ell}, k = 1, \ldots, 4, \ell = 1, \ldots, 3)$, instead of $(4 \times 500 + 3 \times 1000)$ parameters with two Bernoulli mixture models applied on I and J separately.

2.3 Estimation of the Parameters

Now we focus on the estimation of an optimal value of $\boldsymbol{\theta}$ by the maximum likelihood approach associated to this block mixture model. For this model, the

complete data are taken to be the vector $(\mathbf{x}, \mathbf{z}, \mathbf{w})$ where unobservable vectors $\mathbf{z}$ and $\mathbf{w}$ are the labels; the classification log-likelihood

$$L_C(\mathbf{z}, \mathbf{w}, \boldsymbol{\theta}) = L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{z}, \mathbf{w}) = \log f(\mathbf{x}, \mathbf{z}, \mathbf{w}; \boldsymbol{\theta})$$

can then be written

$$L_C(\mathbf{z}, \mathbf{w}, \boldsymbol{\theta}) = \sum_{i,k} z_{ik} \log p_k + \sum_{j,\ell} w_{j\ell} \log q_\ell + \sum_{i,j,k,\ell} z_{ik} w_{j\ell} \log \varphi(x_{ij}; \alpha_{k\ell}).$$

The EM algorithm [4] maximizes the log-likelihood $L_M(\boldsymbol{\theta})$ w. r. to $\boldsymbol{\theta}$ iteratively by maximizing the conditional expectation of the complete data log-likelihood $L_C(\mathbf{z}, \mathbf{w}, \boldsymbol{\theta})$ w. r. to $\boldsymbol{\theta}$ given a previous current estimate $\boldsymbol{\theta}^{(t)}$ and the observed data $\mathbf{x}$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}) = \sum_{i,k} c_{ik}^{(t)} \log p_k + \sum_{j,\ell} d_{j\ell}^{(t)} \log q_\ell + \sum_{i,j,k,\ell} e_{ikj\ell}^{(t)} \log \varphi(x_{ij}; \alpha_{k\ell}), \quad (1)$$

with

$$\begin{aligned} c_{ik}^{(t)} &= P(Z_{ik} = 1 | \boldsymbol{\theta}^{(t)}, \mathbf{X} = \mathbf{x}), \\ d_{j\ell}^{(t)} &= P(W_{j\ell} = 1 | \boldsymbol{\theta}^{(t)}, \mathbf{X} = \mathbf{x}), \\ e_{ikj\ell}^{(t)} &= P(Z_{ik} W_{j\ell} = 1 | \boldsymbol{\theta}^{(t)}, \mathbf{X} = \mathbf{x}), \end{aligned}$$

where the upper case letters $\mathbf{X}$, Z_{ik} and $W_{j\ell}$ denote the random variables.

Unfortunately, difficulties arise owing to the dependence structure among the variables X_{ij} of the model, and more precisely, to the determination of $e_{ikj\ell}^{(t)}$. To solve this problem a variational approximation by the product $c_{ik}^{(t)} d_{j\ell}^{(t)}$ and a use of the Generalized EM algorithm (GEM) provide a good solution in the clustering and estimation contexts [14].

Next we develop the *Generative Topographic Mapping* which is based on a constrained block Bernoulli mixture whose parameters can be optimized by using a Generalized EM algorithm.

3 Block Generative Topographic Mapping

The Generative Topographic Mapping is a method similar to SOM but based on a constrained gaussian mixture density estimation. The clusters are typically arranged in a regular grid, which is the latent discretized space. The parameters are parameterized as a linear combination of g vectors of h smooth nonlinear basis functions ϕ evaluated on g coordinates of a rectangular grid $\{s_k\}_{k=1}^{k=g}$, so for $k = 1, \cdots, g$ we note

$$\xi_k = \Phi(s_k) = (\phi_1(s_k), \phi_2(s_k), \cdots, \phi_h(s_k))^T,$$

where each basis function ϕ is a kernel-like function,

$$\phi(s_k) = exp\Big(- \frac{||s_k - \mu_\phi||^2}{2\nu_\phi^2} \Big),$$

with $\mu_\phi \in \mathbb{R}^2$ a mean center and ν_ϕ a standard deviation. More formally, we parameterize the $\alpha_{k\ell}$'s of the block latent model by using the latent space projected into a higher space of h dimensions and we obtain m new h-dimensional unknown vectors noted w_ℓ to be estimated. To keep the dependence on ℓ and k of $\alpha_{k\ell}$, we use the inner product $w_\ell^T \xi_k$ which is then normalized to a probability by the sigmoid function $\sigma(.)$ as a parameter of the Bernoulli pdf. With this formulation, the $g \times m$ matrix $\boldsymbol{\alpha}$ is replaced by the $h \times m$ matrix $\boldsymbol{\Omega} = [w_1|w_2|\cdots|w_m]$. As h is small in practice, as several tens, the model remains parsimonious. In the previous example where the binary data consists of 1000 rows and 500 columns, we end to about several hundred $h \times m$ parameters because h is typically less than 40 and m less than 10. The number of parameters is still less than in the case of a classical mixture approach applied to the both sets separately. Our model has a good foundation to avoid overfitting and its estimation may be less prone to fall into local optima thanks to the small number of parameters: alternative models have a linear increasing of the number of their parameters when the dimension of the data space becomes higher, contrary to the Block GTM. The following figure 1 shows how the discretized plane becomes a non linear space of probability with the constraints of vicinity.

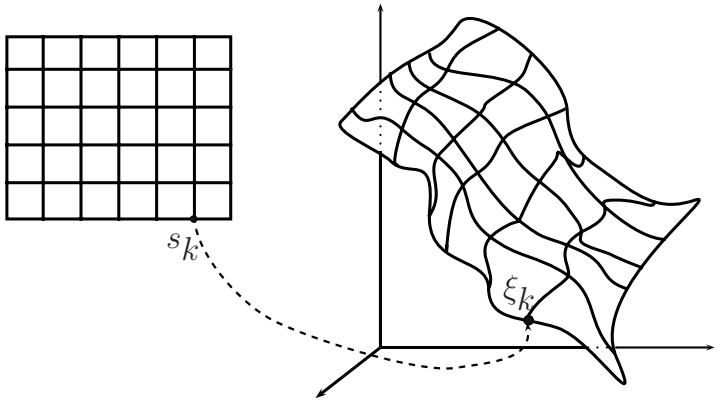

Fig. 1. The graphic illustrates the parameterization of the non linear sigmoid with transformation from a bidimensional Euclidean space to a space of parametric probabilities. In the left the rectangular mesh of the s_k's coordinates is drawn, and in the right the distribution space from φ. Each coordinate of the mesh s_k, $k = 1, \cdots, g$, is mapped in order to become a Bernoulli pdf by writing $\sigma(w_\ell^T \xi_k)$, $\ell = 1, \cdots, m$.

The maximization of the new expression of (1) depending on $\boldsymbol{\Omega}$ can also be performed by the alternated maximization of conditional expectations $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}|\mathbf{d})$ and $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}|\mathbf{c})$ [14]. When the proportions are supposed equal, the two criteria take the following form

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}|\mathbf{d}) = \sum_{i,k} c_{ik}^{(t)} \Big\{ \sum_\ell u_{i\ell} w_\ell^T \xi_k - d_\ell \log(1 + e^{w_\ell^T \xi_k}) \Big\},$$
$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}|\mathbf{c}) = \sum_{j,\ell} d_{j\ell}^{(t)} \Big\{ \sum_k v_{jk} w_\ell^T \xi_k - c_k \log(1 + e^{w_\ell^T \xi_k}) \Big\},$$

with $u_{i\ell}=\sum_j d_{j\ell}^{(t)} x_{ij}$, $d_\ell=\sum_j d_{j\ell}$ and $c_{ik}^{(t)} \propto \prod_\ell (\sigma(w_\ell^T \xi_k))^{u_{i\ell}} (1-\sigma(w_\ell^T \xi_k))^{d_\ell - u_{i\ell}}$, and $v_{jk}=\sum_i c_{ik}^{(t)} x_{ij}$, $c_k = \sum_i c_{ik}$ and $d_{j\ell}^{(t)} \propto \prod_k (\sigma(w_\ell^T \xi_k))^{v_{jk}} (1-\sigma(w_\ell^T \xi_k))^{c_k - v_{jk}}$. A closed form for maximizing these two expectations does not exist yet because of the non linearities from the sigmoid functions, so we use a gradient approach to calculate

$$w^{(t+\frac{1}{2})} = argmax_w \; Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)} | \mathbf{d}) \text{ and } w^{(t+1)} = argmax_w \; Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t+\frac{1}{2})} | \mathbf{c}).$$

By derivative of the two criteria, we get the gradient vectors $\mathbf{Q}_u^{(t)}$, $\mathbf{Q}_v^{(t)}$, and the Hessian matrices $\mathbf{H}_u^{(t)}$, $\mathbf{H}_v^{(t)}$. As the Hessian are block diagonal matrices, we are able to increase the log-likelihood at each step of EM, by two consecutive Newton-Raphson ascents for $\ell = 1, \ldots, m$. This leads to the Generalized EM algorithm. If we note $\Phi = [\xi_1 | \xi_2 | \cdots | \xi_g]^T$ the $g \times h$ matrix of basis functions, each w_ℓ is then expressed as

$$\begin{aligned} w_\ell^{(t+\frac{1}{2})} &= w_\ell^{(t)} + \tfrac{1}{d_{(\ell)}} \Big(\Phi^T G F_\ell \Phi \Big)^{-1} \Big(\Phi^T C u_\ell - d_{(\ell)} \, \Phi^T G \alpha_\ell \Big), \\ w_\ell^{(t+1)} &= w_\ell^{(t+\frac{1}{2})} + \tfrac{1}{d_{(\ell)}} \Big(\Phi^T G F_\ell \Phi \Big)^{-1} \Big(\Phi^T V d_\ell - d_{(\ell)} \, \Phi^T G \alpha_\ell \Big), \end{aligned}$$

where $C = (c_{ik}^{(t)})$ is a $g \times n$ matrix of posterior probabilities, $V = (v_{jk}^{(t)})$ a $g \times d$ matrix of sufficient statistics, $G = (c_k^{(t)})$ and $F_\ell = (\alpha_{k\ell}^{(t)}(1 - \alpha_{k\ell}^{(t)}))$ are $g \times g$ diagonal matrices, $\alpha_\ell = (\alpha_{k\ell}^{(t)})$ a $g \times 1$ vector, $u_\ell = (u_{i\ell}^{(t)})$ a $n \times 1$ vector, $d_\ell = (d_{j\ell}^{(t)})$ a $d \times 1$ vector, and $d_{(\ell)} = d_\ell^{(t)}$ is a scalar.

Finally, for each ℓ, the current parameters $w_\ell^{(t)} \in \mathbb{R}^h$ converges towards the solution $\hat{w}_\ell$. To avoid overfitting and bad numerical solutions, we use a bayesian gaussian prior [16] inducing the bias $-\eta_\ell ||w_\ell||^2/2$ for each w_ℓ. The correction of the estimates is then done by adding $-\eta_\ell w_\ell$ to the gradient and $-\eta_\ell \mathbb{I}_h$ to the diagonal of the Hessian, where $\mathbb{I}_h$ is the h-dimensional identity matrix. The value of the hyperparameters η_ℓ can be manually chosen or estimated.

This Newton-Raphson process in a matrix form sounds like an IRLS [17] step, a crude alternative is a simple gradient with training constant ρ_u and ρ_v instead of the Hessian inverse. Finally, one can notice that the symmetry of the two original mirrored formulas for each side of the matrix is lost because only rows are mapped. Next we illustrate the proposed model on several datasets and demonstrate its good behavior in practice.

4 Numerical Experiments

We experiment our new mapping method on three classical datasets to illustrate the approach. The initialization of the map is done with the help of the first factorial plane from Correspondence Analysis [18], by drawing a mesh over this plane and constructing the initial Bernoulli parameters $\alpha_{k\ell}^{(0)}$ according to this crude clustering.

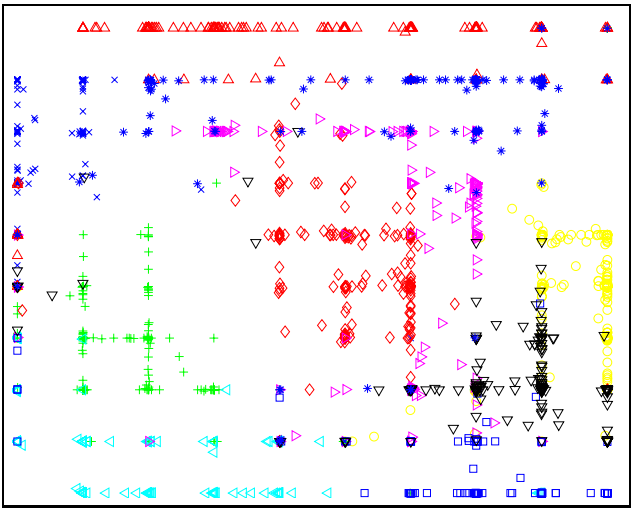

Fig. 2. The Block GTM mapping of the 2000×240 image matrix from binarized digits

The first dataset is compound of 2000 binarized images from a database of handwritten digits. For each of the 10 digits '0', '1' and '9', there are 200 images which were digitalized into 240 multi-dimensional vectors, so the data matrix is 2000×240 with 10 classes for the row side. No information about class for the column side is provided. The mapping of these data is presented in figure 2 which shows quite good separation of the classes, close to that of the early work of [19,20]. We used a map of size 10 by 10, and 9 nonlinear basis functions plus one intercept and the linear position of the node over the plane, so $h = 12$, and $g = 100$. We choose empirically the value of $m = 20$ as a good number of classes for columns after several manual trials. On the figure 2, the posterior means, $\sum_k \hat{c}_{ik} s_k$, are visualized by a different symbol and color plot for each different class label.

To check more easily the block latent model property and the behavior of the proposed algorithm, two textual datasets are studied with $m = 10$, $g = 81$ and $h = 28$.

The second dataset is compound of 400 selected documents from a textual database of 20000 news. Four newsgroups among the twenty existing ones were kept: "sci.cryp", "sci.space", "sci.med", "soc.religion.christian". For each newsgroup, 100 mails were chosen randomly. The data matrix was then constructed as following. From all the texts, the whole vocabulary of the stemmed words is sought for the entire corpus. Then, a first matrix is constructed with its rows corresponding to texts, and columns corresponding to terms. The value of a cell in this matrix is the number of occurrence of the word in the text. The final list of words is chosen by evaluating mutual information to maximize separation between classes of document thanks to known labels. The final matrix is 400×100 with 4 clusters of documents [19,20]. The mapping of this texts on

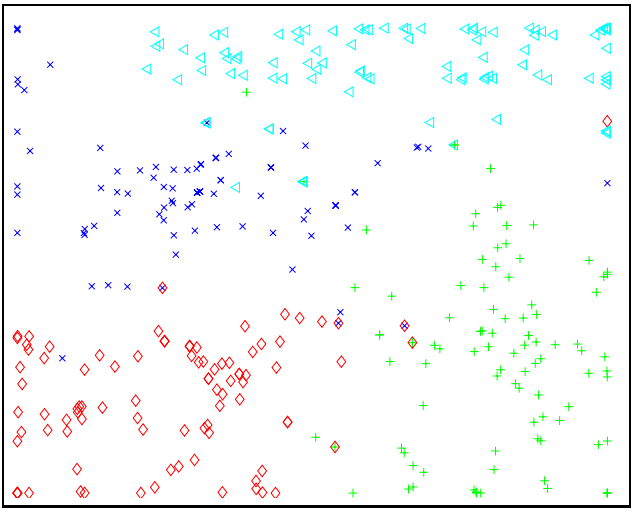

Fig. 3. The Block GTM mapping of the 400×100 textual matrix from four newsgroups

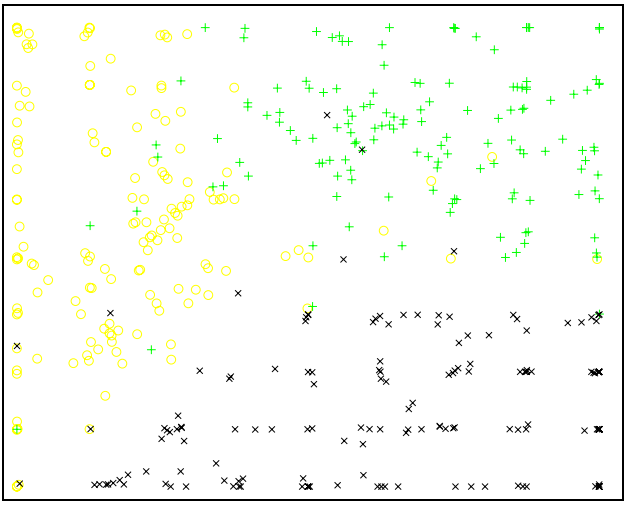

Fig. 4. The Block GTM mapping of the 449×167 textual matrix from three scientific datasets

figure 3 reveals the four topics of discussion which are easily recognizable. The classes are well separated with precise frontiers and we were able to interpret clusters of words too.

The third dataset is a sample of the Classic3 [21] matrix which is a bag of words coding of scientific articles. They come from the three bases Medline, Cisia, Cranfield. We select 450 documents from this file, by randomly drawing

150 documents from each cluster. We select the more frequent words over 30 from all the vocabulary of 4303 terms and we end to a random matrix with approximately 450 rows and 170 columns while discarding the empty rows. One of these matrices was then mapped. Our approach permits us to observe the behavior of the algorithm and we noticed that the solution was stable with only some few outliers badly placed over the plane. The reason is that the clusters are not perfectly separated and that binary coding is not the optimal way to see textual contents. Moreover, we observe quite similar mapping by our binary solution when comparing with the state of art multinomial model-based mapping for the newsgroups file. In future, a contingency table should be modeled to get an even better result in the textual case. Despite this remark, we are still able to visualize the three classes almost perfectly well separated by the non linear mapping in the Figure 4.

5 Conclusion

Considering clustering and visualization within the mixture model approach, we have proposed a new generative self-organizing map for binary data. The proposed Block Generative Topographic Mapping achieves topological organization of the cluster centers basing on a parsimonious block latent model. It counts far fewer parameters than the previously existing models based on a multivariate Bernoulli mixture model [19], a multinomial pLSA [22] or a Bernoulli pLSA [5]. In table 1, when we consider the clustering only on the rows and the proportions p_k and q_ℓ being equal, we report the number of parameters used from the cited models. We note that with our model, the number of parameters increases only with the number of column clusters.

In the visualization context, our variant of GTM gives encouraging results on three applications in two real domains (images and texts). While the linear correspondence analysis is not able to show separately the different clusters over this first plane, our algorithm appears more efficient. Furthermore, the number of parameters of an alternative of the multinomial model in [23,24] for binary data remains the same as for the unconstrained model. So the Block GTM appears clearly as the best candidate to scale for data mining problems.

A first appealing perspective of the model is in domain of textual analysis. Thanks to the clustering of the columns, we are able to map clusters for texts

Table 1. The number of parameters for Bernoulli probabilistic SOMs with $m \ll d$

Model	unconstrained	constrained
Bernoulli mixture model	gd	hd
Bernoulli pLSA	$(n+d)g$	$ng+dh$
Multinomial pLSA	gd	hd
Block latent model	gm	hm

and words together, by evaluating the new heuristic probability that the j-st word belongs to the k-st class, with the following formula

$$d_{jk} \propto \sum_{\ell} d_{j\ell}\alpha_{k\ell}$$

which appears as a crude marginalization over an hidden random variable classifying the columns. The first experiments provide promising results. The vocabulary from each topic appears clearly more probable where each corresponding topic lies on the map. This clustering is very different from the usual one: in the literature, it is usually shown the most probable terms for each cluster of document. A distribution over the map is learned for rows and indirectly for columns, so an original perspective is to construct a non linear *biplot* as [25] by a fully probabilistic and automatic method.

References

1. Lebart, L., Morineau, A., Warwick, K.: Multivariate Descriptive Statistical Analysis. J. Wiley, Chichester (1984)
2. Kohonen, T.: Self-organizing maps. Springer, Heidelberg (1997)
3. Bishop, C.M., Svensén, M., Williams, C.K.I.: Developpements of generative topographic mapping. Neurocomputing 21, 203–224 (1998)
4. Dempster, A., Laird, N., Rubin, D.: Maximum-likelihood from incomplete data via the EM algorithm. J. Royal Statist. Soc. Ser. B. 39, 1–38 (1977)
5. Priam, R., Nadif, M.: Carte auto-organisatrice probabiliste sur données binaires (in french). In: RNTI (EGC 2006 proceedings), pp. 445–456 (2006)
6. Bock, H.: Simultaneous clustering of objects and variables. In: Diday, E. (ed.) Analyse des Données et Informatique, INRIA, pp. 187–203 (1979)
7. Govaert, G.: Classification croisée. In: Thèse d'état, Université Paris 6, France (1983)
8. Govaert, G.: Simultaneous clustering of rows and columns. Control and Cybernetics 24(4), 437–458 (1995)
9. Cottrell, M., Ibbou, S., Letrémy, P.: Som-based algorithms for qualitative variables. Neural Networks 17(89), 1149–1167 (2004)
10. Symons, M.J.: Clustering criteria and multivariate normal mixture. Biometrics 37, 35–43 (1981)
11. McLachlan, G.J., Basford, K.E.: Mixture Models, Inference and applications to clustering. Marcel Dekker, New York (1988)
12. McLachlan, G.J., Peel, D.: Finite Mixture Models. John Wiley and Sons, New York (2000)
13. Govaert, G., Nadif, M.: Clustering with block mixture models. Pattern Recognition 36, 463–473 (2003)
14. Govaert, G., Nadif, M.: An EM algorithm for the block mixture model. IEEE Trans. Pattern Anal. Mach. Intell. 27(4), 643–647 (2005)
15. Govaert, G., Nadif, M.: Block clustering with bernoulli mixture models: Comparison of different approaches. Computational Statistics & Data Analysis 52, 3233–3245 (2008)
16. MacKay, D.J.C.: Bayesian interpolation. Neural Computation 4(3), 415–447 (1992)

17. McCullagh, P., Nelder, J.: Generalized linear models. Chapman and Hall, London (1983)
18. Benzecri, J.P.: Correspondence Analysis Handbook. Dekker, New-York (1992)
19. Girolami, M.: The topographic organization and visualization of binary data using multivariate-bernoulli latent variable models. IEEE Transactions on Neural Networks 20(6), 1367–1374 (2001)
20. Kabán, A., Girolami, M.: A combined latent class and trait model for analysis and visualisation of discrete data. IEEE Trans. Pattern Anal. and Mach. Intell., 859–872 (2001)
21. Dhillon, I.: Co-clustering documents and words using bipartite spectral graph partitioning. In: Seventh ACM SIGKDD Conference, San Francisco, California, USA, pp. 269–274 (2001)
22. Hofmann, T.: Probmap - a probabilistic approach for mapping large document collections. Intell. Data Anal. 4(2), 149–164 (2000)
23. Kaban, A.: A scalable generative topographic mapping for sparse data sequences. In: ITCC 2005: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC 2005), Washington, DC, USA, vol. I, pp. 51–56. IEEE Computer Society, Los Alamitos (2005)
24. Kaban, A.: Predictive modelling of heterogeneous sequence collections by topographic ordering of histories. Machine Learning 68(1), 63–95 (2007)
25. Priam, R.: CASOM: Som for contingency tables and biplot. In: 5th Workshop on Self-Organizing Maps (WSOM 2005), pp. 379–385 (2005)